Covert Channel Synthesis for Transducers

Gilles Benattar, Béatrice Bérard, Didier Lime, John Mullins, Olivier Henri Roux, Mathieu Sassolas

Covert Channel Synthesis for Transducers*

Gilles Benattar[1], Béatrice Bérard[2], Didier Lime[3], John Mullins[4,**], Olivier H. Roux[1], and Mathieu Sassolas[2]

[1] Université de Nantes, IRCCyN, CNRS UMR 6597, Nantes, France
[2] Université Pierre & Marie Curie, LIP6/MoVe, CNRS UMR 7606, Paris, France
[3] École Centrale de Nantes, IRCCyN, CNRS UMR 6597, Nantes, France
[4] École Polytechnique de Montréal, Montréal (Québec), Canada.

Email: Gilles.Benattar@irccyn.ec-nantes.fr, mathieu.sassolas@lip6.fr

Abstract. Covert channels are a security threat for information systems, since they permit illegal flows, and sometimes leaks, of classified data. Although numerous descriptions have been given at a concrete level, relatively little work has been carried out at a more abstract level, outside probabilistic models. In this paper, we propose a definition of covert channels based on encoding and decoding binary messages with transducers, in a finite transition system. We first compare this notion of covert channel with a similar one called iterated interference. We then give a necessary condition for the existence of a covert channel. Unfortunately, in the general case of our setting, it turns out that the existence of a covert channel is undecidable. However, restricting to functional systems, we obtain a PTIME procedure to decide the existence of a covert channel.

Keyword: Security, Covert Channels, Non-interference, Transducers.

1 Introduction

1.1 General context and related work

The context of this work is the security of information flows. While systems have to communicate to exchange information and share resources, they aim at maintaining some confidentiality and try to establish security levels to forbid or filter information flows, preventing leaks of classified data. A covert channel is a way to bypass system securities in order to recover some confidential information. Well-known examples are described in [1] for TCP/IP, in which reserved fields of IP packets were used to transmit information. Characteristics such as running time [2],

* Work partially supported by project DOTS (ANR-06-SETI-003)
** Author partially supported by an NSERC discovery grant (Government of Canada)

power consumption [3] and even electromagnetic radiation [4] have also been exploited to recover confidential information from different security systems.

Since their introduction by Lampson [5], covert channels have been largely studied in terms of security policies *à la* Bell and La Padula [6]. But access control does not provide complete solutions for protecting information and as a complementary approach, non-interference was introduced in [7] to detect covert channels through the information flow of a multi-level security system in order to prevent high-level data from being deduced by low-level parties. This work has been extended in [8] for CCS processes. Many behavioral equivalences have been considered in order to establish a wide variety of non-interference properties classified according to their discrimination power.

However, as explained in [9], non interference is too strong a requirement since a system fails to satisfy non-interference as soon as it leaks only one bit of information. Thus, quantifying information leak is necessary. Moreover, in [10], a zero capacity channel is given, on which any message can be sent. This observation has led to set out an additional condition, called the *small message criterion*, to the existence of a covert channel. It roughly states that messages of arbitrary size can be sent in finite time. Many models of covert channels have been proposed, based on information-theoretic metrics to measure information revealed to an attacker [9,11,12,13].

Another research thread focuses on a different approach by re-formulating qualitative information flow policies [14,15,16] in order to cope with the above discussed limitations of the original condition. For instance, opacity [17] is a more general notion where different observation functions are compared with respect to their power of discovering secret (or opaque) information. While opacity is undecidable, some positive results were obtained in [18,19] for unbounded Petri nets and finite transition systems, and in [20] for computing optimal control of a system enforcing concurrent secrets. In [21], the authors describe covert channels as *iterated interference* based on observations from [9,10]. They consider systems modeled as hierarchical message sequence charts and transformed into Büchi games, with transducers for encoding and decoding messages of arbitrary size. In this setting, the existence of an effective covert channel corresponds to the existence of a strategy and is proved decidable, under certain restrictions for the model and the transducers.

1.2 Contribution.

In this work, we follow the latter qualitative approach and we propose a general definition for covert channels, in the framework of rational transducers. There is a potential covert channel if we can find a way to encode and decode any binary message, and if the encoder and decoder mechanisms, defined as transducers, can be computed. We show that this notion of covert channel is different from interference and iterated interference. The problem of covert channel detection is then to synthesize the two encoding and decoding transducers. We give a necessary condition for existence of a covert channel: this condition consists in the presence of what we call an *encoding* state, a condition which was directly considered as part of the covert channel definition in [21]. We also show that the existence problem is undecidable in the general case, but can be solved in polynomial time when the system is functional.

1.3 Outline.

Section 2 gives preliminary definitions and Section 3 compares the notions of covert channels and iterated interference. Section 4 shows how to reduce the existence of an effective covert channel to a simpler problem, yielding a necessary condition on the existence of a covert channel. Section 5 proves that the problem of existence of a cover channel is undecidable in the general case, while Section 6 provides a procedure for decision in the case of functional transducers. We discuss open problems and future work in Section 7.

2 Preliminaries

In this section, we recall general definitions used in the sequel.

2.1 Finite words

Let A be a finite alphabet and A^* the set of words over A, with ε for the empty word. A *language* is a subset of A^* and we set $A^\varepsilon = A \cup \{\varepsilon\}$. The length of a word u is written $|u|$ and for $1 \leq i \leq |u|$, $u[i]$ is the ith letter of u. If B is a subset of A, then $|u|_B = |\{i \in \mathbb{N} \mid u[i] \in B\}|$ is the number of letters of u that are in B. The *projection* on B^*, denoted by $\mathbf{proj}_B$, is the morphism from A^* onto B^* such that $\mathbf{proj}_B(a) = a$ if $a \in B$ and ε if $a \in A \setminus B$.

We recall a simple form of the folklore *Defect Theorem* ([22] or [23]):

Lemma 1. *Let $u, v \in A^*$ be two words. Then $uv = vu$ if and only if there exist a word w and two integers $m, n \geq 0$ such that $u = w^m$ and $v = w^n$.*

As a consequence, it can be proved that if u and v do not commute, the set $\{u, v\}$ is a *code*: any word x in the language $\{u, v\}^*$ has a unique decomposition

$$x = w_1 \cdots w_n \quad \text{where} \quad \forall 1 \leq i \leq n,\ w_i \in \{u, v\}$$

Moreover, given word u, there exists a word w such that the set of words that commute with u is w^*.

For two words u and v, we write $v \preccurlyeq u$ when v is a *prefix* of u. We say that v is a k-bounded prefix of u if its length differs from the length of u by at most k letters and we denote by $\mathit{Pref}_k(u)$ the set of k-bounded prefixes of u: $\mathit{Pref}_k(u) = \{v \in A^* | v \preccurlyeq u \wedge |u| - |v| \leq k\}$. For instance, over $A = \{0, 1\}$, $\mathit{Pref}_2(010110) = \{010110, 01011, 0101\}$.

2.2 Transition Systems

A *labeled transition system* (LTS) is a tuple $\mathcal{M} = \langle S, s_0, \mathit{Lab}, \Delta \rangle$ where S is a set of states, s_0 is the initial state, Lab is a set of labels and $\Delta \subseteq S \times \mathit{Lab} \times S$ is the transition relation. We use the notation $s \xrightarrow{a} s'$ if $(s, a, s') \in \Delta$. An LTS $\mathcal{M}$ is finite if S and Δ are finite. A run from $s \in S$ is a finite sequence of transitions $\rho = s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} s_n$. The last state of the sequence, *i.e.* the state s_n, is denoted by $\mathit{last}(\rho)$ and the *trace* of ρ is $\mathit{trace}(\rho) = a_1 \cdots a_n$. We write $s \overset{w}{\Rightarrow} s'$ if there is a run ρ from s to s' with trace w. The set of runs starting from s in $\mathcal{M}$ is $\mathit{Runs}(s, \mathcal{M})$ and $\mathit{Runs}(\mathcal{M}) = \mathit{Runs}(s_0, \mathcal{M})$.

A finite automaton, or automaton for short, $\mathcal{M} = \langle S, s_0, \mathit{Lab}, \Delta, F \rangle$, is a finite LTS along with a set $F \subseteq S$ of final states. A word $w \in \mathit{Lab}^*$ is *accepted* by $\mathcal{M}$ if $w = \mathit{trace}(\rho)$ for some $\rho \in \mathit{Runs}(\mathcal{M})$ with $\mathit{last}(\rho) \in F$. The language accepted by $\mathcal{M}$, denoted by $\mathcal{L}(\mathcal{M})$, is the set of words accepted by $\mathcal{M}$. A rational language is a language accepted by a finite automaton. Note that the languages considered here are often *prefix-closed*. When the set of final states is omitted, it implicitly means that $F = S$ *i.e.* all states are final states. We define the sets $\mathit{Reach}(\mathcal{M}) = \{s \in S \mid \exists \rho \in \mathit{Runs}(\mathcal{M}), \mathit{last}(\rho) = s\}$ and $\mathit{CoReach}(\mathcal{M}) = \{s \in S \mid \exists \rho \in \mathit{Runs}(s, \mathcal{M}), \mathit{last}(\rho) \in F\}$. If a state $s \in S$ does not belong to $\mathit{Reach}(\mathcal{M}) \cap \mathit{CoReach}(\mathcal{M})$, then $\mathcal{M}' = \langle S \backslash \{s\}, s_0, \mathit{Lab}, \Delta, F \rangle$ accepts the same language as $\mathcal{M}$. Therefore, in the sequel, we only consider automata for which $S = \mathit{Reach}(\mathcal{M}) \cap \mathit{CoReach}(\mathcal{M})$.

2.3 Rational Transducers

A relation τ between two sets E and F is a subset of $E \times F$. For $e \in E$, the set of images of e by τ is $\tau(e) = \{f \in F \mid (e, f) \in \tau\}$. The domain of τ is $Dom(\tau) = \{e \in E \mid \exists f \in F, (e, f) \in \tau\}$ and the image of τ is $Im(\tau) = \{f \in F \mid \exists e \in E, (e, f) \in \tau\}$.

For an alphabet A, and a subset P of A^*, we denote by $Id(P)$ the identity relation $\{(w, w) \mid w \in P\}$ on $A^* \times A^*$ and by $Id_k(P)$ the relation between words and their k-bounded prefixes in P: $Id_k(P) = \{(u, v) \in P \times P \mid v \in Pref_k(u)\}$. Note that $Id_0 = Id$.

Given alphabets A and B, a rational transducer (or transducer for short) is a finite automaton whose set of labels is $A^* \times B^*$. The language accepted by a transducer $\mathcal{M}$ is a rational relation [24] between A^* and B^*. The transducer $\mathcal{M}$ is said to implement the corresponding relation which is also denoted by $\mathcal{M}$. Hence, the set of images by $\mathcal{M}$ of a word $w \in A^*$ is written $\mathcal{M}(w)$. When $\mathcal{M}(w)$ is a singleton, it will also denote its only element, with a slight abuse of notations. If the domain of $\mathcal{M}$ is A^*, then $\mathcal{M}$ is said to be *complete*. The transducer is *functional* if for each word $w \in A^*$, there is at most one word in $\mathcal{M}(w)$. The composition of rational transducers $\mathcal{M}$ on $A^* \times B^*$ and $\mathcal{M}'$ on $B^* \times C^*$, denoted by $\mathcal{M}' \circ \mathcal{M}$, is a rational transducer on $A^* \times C^*$, as shown by Elgot and Mezei in [25]. Moreover, the image and the inverse image of a rational set by a rational transducer is rational [24].

Any transducer on $A^* \times B^*$ admits a *normal form* where each transition is labeled either by (a, ε), also written $a|\varepsilon$, or by (ε, b), written $\varepsilon|b$, with $a \in A^\varepsilon$ and $b \in B^\varepsilon$. This representation preserves the accepted relation and can be used to syntactically derive an automaton on alphabet $A \uplus B \cup \{\varepsilon\}$ from a transducer, where $\uplus$ denotes disjoint union: if $\mathcal{M} = \langle S, s_0, A^* \times B^*, \Delta, F\rangle$ is transducer in normal form, the automaton $\mathcal{M}' = (S, s_0, A \uplus B \cup \{\varepsilon\}, \Delta', F)$ is obtained by defining Δ' as follows:

- If $s \xrightarrow{h|\varepsilon} s' \in \Delta$, then $s \xrightarrow{h} s' \in \Delta'$
- If $s \xrightarrow{\varepsilon|\ell} s' \in \Delta$, then $s \xrightarrow{\ell} s' \in \Delta'$
- If $s \xrightarrow{\varepsilon|\varepsilon} s' \in \Delta$, then $s \xrightarrow{\varepsilon} s' \in \Delta'$

3 Opacity, Iterated Interference, and Covert Channels

We now define a transducer based model for covert channels and compare this definition with interference, a notion sometimes considered too tight to effectively test real-world security policies [16,26].

We consider a system with two users: a high-level user whose actions are in an alphabet H and a low-level user whose actions are in alphabet L. In our setting, each user can only execute and see its own actions and the system will be described by a transducer $\mathcal{M}$ implementing a rational relation of $H^* \times L^*$. The system contains a covert channel if high-level actions can influence low-level ones in a way to ensure that any binary message can be transmitted. Hence, the transducer model abstracts the system as a black box reading inputs on alphabet H and producing outputs on L. The only restriction put so far on $\mathcal{M}$ is that the relation between input and output is a rational relation.

On the other hand, we also assume that the mechanism used to transmit a message through $\mathcal{M}$ is limited by defining an encoder $\mathcal{E}$ as a transducer that reads binary input and produces output in H. Symmetrically, the decoding mechanism $\mathcal{D}$ is also a rational transducer which decodes letters in L into a binary word.

3.1 Transducer model of a covert channel

The definition below states that there is a covert channel if the message is correctly transmitted. However, to take into account possible delays of transmission, we do not require that the binary word obtained by $\mathcal{D}$ be strictly identical to the word initially sent by $\mathcal{E}$. Rather, we accept as a result a k-bounded prefix, for some $k \geq 0$.

Definition 1. *Two transducers $\mathcal{E}$ on $\{0,1\}^* \times H^*$ and $\mathcal{D}$ on $L^* \times \{0,1\}^*$* implement *a covert channel of delay k for a transducer $\mathcal{M} = \langle S, s_0, H^* \times L^*, \Delta, F\rangle$ if*

$$Id(\{0,1\}^*) \subseteq \mathcal{D} \circ \mathcal{M} \circ \mathcal{E} \subseteq Id_k(\{0,1\}^*).$$

Note that since all binary words must be encoded, existence of a covert channel implies that encoder $\mathcal{E}$ is complete. A system containing a covert channel is described in the following example, adapted from [21].

Example 1. We consider a simple transmission medium in which packets can be either long or short. It is assumed that the content of the packets themselves are monitored and therefore no sensitive information can be transmitted by this means. The higher level user asks the medium to open a connection, then transmits either a long or a short packet. A short packet is transmitted in one step while a long packet is transmitted in two parts by the medium: an incomplete packet followed by a completed one, the type of the latter being a short packet type. The corresponding

transducer is depicted in Fig. 1. A way to code any binary message is to transmit a long message for a 0 and a short one for a 1. The receiver decodes a sequence of an incomplete packet followed by a completed one by a 0, and a complete packet by a 1. The corresponding encoder and decoder are displayed in Fig. 2.

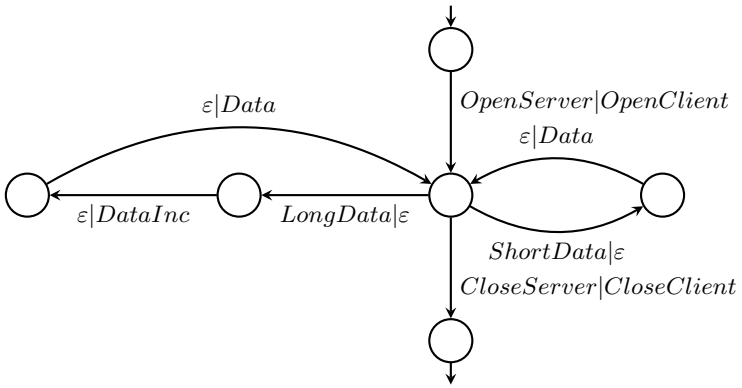
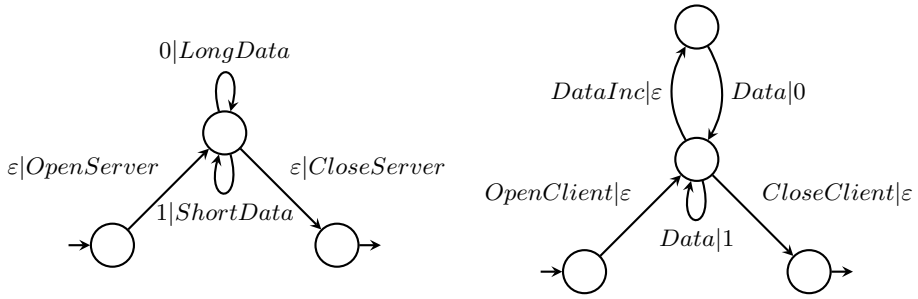


Fig. 1. Transducer $\mathcal{M}_1$ for packet transmission medium



(a) $\mathcal{E}_1$ is an encoder for $\mathcal{M}_1$ (b) $\mathcal{D}_1$ is a decoder for $\mathcal{M}_1$

Fig. 2. Implementation of a covert channel of delay 0 for $\mathcal{M}_1$ of Fig. 1

3.2 Comparison with opacity and interference

A system is said to be interferent if an exterior observer can deduce whether an internal (protected) action has occurred [7,8]. Thus, it can be

seen as a one bit leak of information. If an observer can deduce several occurrences of internal actions, the interference is repeated, or *iterated*. We formally define this notion through the more general one of *opacity* [19], which captures a wide range of security properties. We present a simplified version of opacity from [20].

Definition 2 (Opacity). *Let $\mathcal{S}$ and $\mathcal{K}$ be two rational languages over an alphabet A, with $\mathcal{S} \subseteq \mathcal{K}$, and let L be a subset of A. Then $\mathcal{S}$ is said* opaque *with respect to $\mathcal{K}$ if $\mathbf{proj}_L(\mathcal{S}) \subseteq \mathbf{proj}_L(\mathcal{K} \setminus \mathcal{S})$,* i.e. *$\forall w \in \mathcal{S}$, $\exists w' \in \mathcal{K} \setminus \mathcal{S}$ such that $\mathbf{proj}_L(w) = \mathbf{proj}_L(w')$.*

Thus, $\mathcal{S}$ is opaque if low-level observations viewed as words over L cannot distinguish the secrets in $\mathcal{S}$. For a system described by an automaton, non interference states that a low-level user cannot discover if a high-level action occurs. Thus non interference can be expressed as an opacity property where the secrets are words containing at least one high-level letter, hence belong to the language $\mathcal{S}_1 = L^*HA^*$.

Definition 3 (Interference). *Let $\mathcal{S}_k = (L^* \cdot H)^k \cdot A^*$, for any integer $k > 0$. An automaton $\mathcal{M}$ is interferent if $\mathcal{S}_1$ is not opaque w.r.t $\mathcal{L}(\mathcal{M})$. This automaton has an iterated interference if $\forall k > 0$, $\mathcal{S}_k$ is not opaque w.r.t $\mathcal{L}(\mathcal{M})$.*

Covert channels have often been linked to interference or iterated interference. Indeed, a system with no iterated interference will not have a covert channel. However, despite the intuition that iterated interference can yield any number of bits of information to the observer, a system can have iterated interference without having a covert channel. The following example is one such case.

Example 2. Consider the transducer $\mathcal{M}_2$ of Fig. 3, where h is a high-level action and ℓ is a low-level action. We can see that there is an iterated interference, since every time an ℓ is seen, the low-level user knows that there has been at least an h. More formally, if $\mathcal{A}_2$ is the automaton associated with $\mathcal{M}_2$, its language is $\mathcal{L}(\mathcal{A}_2) = (h^+ \cdot \ell)^*$. For an integer $k > 0$, let $w = (h \cdot \ell)^k$. Let $w' \in \mathcal{L}(\mathcal{A}_2) \setminus \mathcal{S}_k$. Since $\mathcal{S}_k$ is the set of words that contain at least k occurrences of h, w' contains at most $k-1$ letters h. As a word of $\mathcal{L}(\mathcal{A}_2)$, w' contains more hs than ℓs, hence w' contains a number $p \leq k-1$ letters ℓ. The projection on $L = \{\ell\}$ of w is ℓ^k while the projection of w' is ℓ^p, with $p < k$.

However, it is impossible for the high-level to encode arbitrary messages for the-low level. This claim is proved in Section 4.2. The underlying

intuition is that any number of h will result in an inferior number of ℓ. This will introduce confusion in any coding, and prevent using the system as a covert channel.

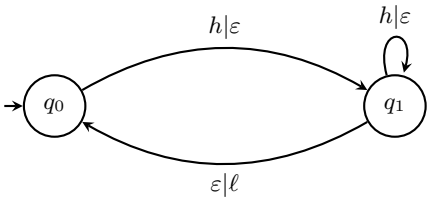


Fig. 3. A transducer $\mathcal{M}_2$ with iterated interference but no covert channel

4 A Necessary Condition for Covert Channels

Our main objective is, given a system, to decide the existence of a covert channel. In our framework, the problem is stated as follows: for a transducer $\mathcal{M}$ from H to L, are there an integer k and two transducers $\mathcal{E}$ on $\{0,1\}^* \times H^*$ and $\mathcal{D}$ on $L^* \times \{0,1\}^*$, that implement a covert channel of delay k on $\mathcal{M}$. In this section, we give a necessary condition for a positive answer to this question. After eliminating the delay parameter, we prove that existence of a covert channel implies the presence of at least one *encoding* node in the system, a condition which was in fact taken as part of the covert channel definition in [21], together with a winning strategy.

4.1 Elimination of the delay parameter

We first show that verifying encoding and decoding can be done for channels of delay 0. Then we show that these channels encompass all channels.

Lemma 2. *Let $\mathcal{M}$ be a transducer on $H^* \times L^*$ and let $\mathcal{E}$ and $\mathcal{D}$ be two transducers on $\{0,1\}^* \times H^*$ and $L^* \times \{0,1\}^*$, respectively. It can be decided whether $\mathcal{E}$ and $\mathcal{D}$ implement a covert channel of delay 0 for $\mathcal{M}$.*

Proof. It can be decided whether a transducer is functional. Moreover, the equality of languages is decidable for functional transducers. In particular, the relation $Id_0(\{0,1\}^*)$ is functional. If $\mathcal{D} \circ \mathcal{M} \circ \mathcal{E}$ is not functional, $\mathcal{E}$ and $\mathcal{D}$ do not implement a covert channel of delay 0. If $\mathcal{D} \circ \mathcal{M} \circ \mathcal{E}$ is functional, it can be decided whether it is equal to the identity, and therefore if $\mathcal{E}$ and $\mathcal{D}$ implement a cover channel of delay 0.

However, deciding whether two transducers implement a covert channel of delay $k \neq 0$ is not straightforward.

Not being able to check candidate encoder and decoder transducers makes the problem of existence of solutions more difficult. Nevertheless, it can be shown that looking only for covert channels with no delay is sufficient.

Lemma 3. *If a transducer $\mathcal{M}$ contains a covert channel of delay k, then it also contains a covert channel with no delay.*

Proof. Suppose $\mathcal{M}$ has a covert channel of delay k. We define the k-repetition morphism $\mathcal{R}_k^\times$ over $\{0,1\}^*$ by:

$$\mathcal{R}_k^\times(0) = 0^{k+1} \text{ and } \mathcal{R}_k^\times(1) = 1^{k+1}$$

In the reverse way, let $\mathcal{R}_k^\div$ be the relation defined by $\mathcal{R}_k^\div = (\mathcal{R}_k^\times)^{-1}$. We claim that $\mathcal{R}_k^\div \circ Id_k(\{0,1\}^*) \circ \mathcal{R}_k^\times = Id(\{0,1\}^*)$. Relations $\mathcal{R}_k^\times$ and $\mathcal{R}_k^\div$ are implemented by the transducers represented in Fig. 4(a) and Fig. 4(b), respectively.

Let $u \in \{0,1\}^*$ and v be its image by $\mathcal{R}_k^\times$. If $u = u_1 \cdots u_n$ with $u_i \in \{0,1\}$, then $v = u_1^{k+1} \cdots u_n^{k+1}$. The k-bounded prefixes of v are $v_0 = u_1^{k+1} \cdots u_n$, $v_1 = u_1^{k+1} \cdots u_n^2, \ldots, v_k = u_1^{k+1} \cdots u_n^{k+1}$. For every $i \in \{0, \ldots, k-1\}$, $\mathcal{R}_k^\div(v_i) = \emptyset$. The image of v_k by $\mathcal{R}_k^\div$ is $u_1 \cdots u_n = u$, so $\mathcal{R}_k^\div \circ Id_k(\{0,1\}^*) \circ \mathcal{R}_k^\times = Id(\{0,1\}^*)$.

Now if $\mathcal{E}$ and $\mathcal{D}$ are two transducers such that $Id(\{0,1\}^*) \subseteq \mathcal{D} \circ \mathcal{M} \circ \mathcal{E} \subseteq Id_k(\{0,1\}^*)$ then $\mathcal{R}_k^\div \circ \mathcal{D} \circ \mathcal{M} \circ \mathcal{E} \circ \mathcal{R}_k^\times \subseteq \mathcal{R}_k^\div \circ Id_k(\{0,1\}^*) \circ \mathcal{R}_k^\times = Id(\{0,1\}^*)$. By taking $\mathcal{E} \circ \mathcal{R}_k^\times$ as an encoder and $\mathcal{R}_k^\div \circ \mathcal{D}$ as a decoder, we obtain a covert channel without delay.

(a) Transducer implementing $\mathcal{R}_k^\times$ (b) Transducer implementing $\mathcal{R}_k^\div$

Fig. 4. Transducers used to suppress the delay in a covert channel

4.2 A general structure of the encoding/decoding pair

Even if it can be decided if two transducers implement a covert channel with no delay, finding those two transducers is hard. In fact, we show in the next section that the general problem is undecidable. However, we reduce the problem by looking only for transducers that have a specific structure. Theorem 1 gives this specific shape depicted in Fig.s 6(a) and 6(b). The proof of this theorem relies on Lemma 4. This lemma exhibits a generic path in an encoding transducer, represented in Fig. 5, where state r accepts two different input words in a loop. From this lemma, an encoding state is obtained for the system itself.

Lemma 4. *Let $\mathcal{A} = \langle S, s_0, M, \Delta, F\rangle$ be an automaton over a (not necessarily free) monoid $(M, \cdot, \varepsilon)$. Suppose there exist $a, a_0, a_1, a' \in M$ such that $a_0 \cdot a_1 \neq a_1.a_0$ and all words of $a\cdot\{a_0, a_1\}^*\cdot a'$ are accepted by $\mathcal{A}$. Then there exist some states $r \in S$ and $f \in F$ and some words $w \in a\cdot\{a_0, a_1\}^*$, $w_0, w_1 \in \{a_0, a_1\}^*$, and $w' \in \{a_0, a_1\}^*\cdot a'$, such that $w_0 \cdot w_1 \neq w_1 \cdot w_0$ and $s_0 \overset{w}{\Longrightarrow} r$, $r \overset{w'}{\Longrightarrow} f$, $r \overset{w_0}{\Longrightarrow} r$ and $r \overset{w_1}{\Longrightarrow} r$.*

Proof. Let $N_{\mathcal{A}}$ be the number of states of $\mathcal{A}$. A simple loop is a run $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} s_n$ with $s = s_n$ and all other states are distinct. In this proof, only simple loops are considered. There are at most $N_{\mathcal{A}} \times 2^{N_{\mathcal{A}}}$ loops in $\mathcal{A}$. Since $\forall m > N_{\mathcal{A}}$, $a \cdot a_0^m \cdot a' \in \mathcal{L}(\mathcal{A})$, there is at least one loop in the automaton whose trace is in $\{a_0\}^*$. Let k_0 be the number of loops in the automaton whose traces are in $\{a_0\}^*$ (or a_0-loop for short). Let $m > N_{\mathcal{A}}$. Consider the words $z = a_0^m \cdot a_1$ and $Z = a \cdot z^{k_0+1} \cdot a'$ and the run

$$\rho = s_0 \overset{a}{\Rightarrow} s_a \overset{a_0^m}{\Longrightarrow} s_1 \overset{a_1}{\Rightarrow} s'_1 \cdots \overset{a_0^m}{\Longrightarrow} s_{k_0+1} \overset{a_1}{\Rightarrow} s'_{k_0+1} \overset{a'}{\Rightarrow} s_{a'}$$

with trace Z, where $s_{a'} \in F$. For $1 \leq j \leq k_0$, let $\rho_j = s'_j \overset{a_0^m}{\Longrightarrow} s_{j+1}$ be the corresponding sub-run of ρ, while ρ_0 is the sub-run $s_a \overset{a_0^m}{\Longrightarrow} s_1$. Each ρ_j contains a a_0-loop. Since there are $k_0 + 1$ such sub-runs, a same a_0-loop of trace $w_0 \in \{a_0\}^*$ occurs in two different sub-runs ρ_{j_1} and ρ_{j_2}. Then the run ρ can be written as

$$s_0 \overset{a}{\Rightarrow} \cdots s_{j_1} \overset{u}{\Rightarrow} r \overset{w_0}{\Longrightarrow} r \overset{v}{\Rightarrow} s'_{j_1+1} \cdots s_{j_2} \overset{u}{\Rightarrow} r \overset{w_0}{\Longrightarrow} r \overset{v}{\Rightarrow} s'_{j_2+1} \cdots s'_{k_0+1} \overset{a'}{\Rightarrow} s_{a'}.$$

Let $w = a \cdot z^{j_1-1} \cdot u$, $w_1 = v \cdot z^{j_2-j_1-1} \cdot u$, and $w' = v \cdot z^{k_0-j_2} \cdot a'$ be the words labeling respectively the sub-run from s_0 to r, the sub-run $r \overset{v}{\Rightarrow} s'_{j_1+1} \cdots s_{j_2} \overset{u}{\Rightarrow} r$, and the sub run from s_{j_2+1} to $s_{a'}$. Then $\mathcal{A}$ contains the runs $s_0 \overset{w}{\Rightarrow} r$, $r \overset{w_0}{\Longrightarrow} r$, $r \overset{w_1}{\Longrightarrow} r$, and $r \overset{w'}{\Longrightarrow} s_{a'}$ where $s_{a'} \in F$, as depicted

in Fig. 5. Moreover, since $u \cdot w_0 \cdot v = a_0^m \cdot a_1$ and $w_0 \in \{a_0\}^*$, v contains a letter a_1; so w_1 contains at least on a_1. Therefore $w_0 \cdot w_1 \neq w_1 \cdot w_0$.

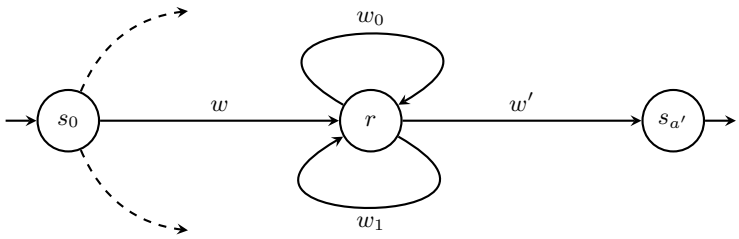


Fig. 5. A generic path in $\mathcal{M}$

This lemma can be instantiated in particular in the case on transducers complete on their input.

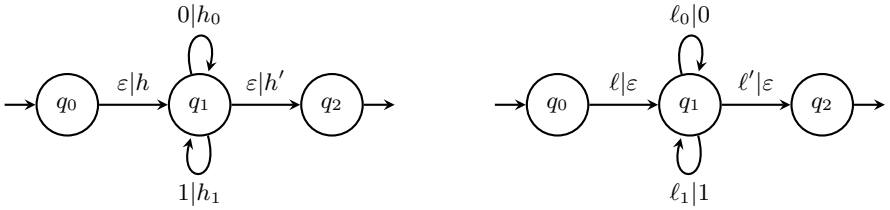
Lemma 5. *Let $\mathcal{E} = \langle S, s_0, \{0,1\}^* \times H^*, \Delta, F\rangle$ be a transducer, complete on its input. Then there exist words $b, b_0, b_1, b' \in \{0,1\}^*$, $h, h_0, h_1, h' \in H^*$, a state $r \in S$ and a state $f \in F$ such that $b_0 \cdot b_1 \neq b_1 \cdot b_0$, $s_0 \overset{b|h}{\Longrightarrow} r$, $r \overset{b'|h'}{\Longrightarrow} f$, $r \overset{b_0|h_1}{\Longrightarrow} r$ and $r \overset{b_1|h_2}{\Longrightarrow} r$.*

Proof. Lemma 4 is applied to the projection of $\mathcal{E}$ on its input.

A pattern having been isolated in the system, we can prove the main theorem of this section. It states that the encoder can be reduced to this very pattern and, by transforming the decoder accordingly, still have a covert channel. More precisely, if two words forming a code can be transmitted by the channel, then it is sufficient to encode 0 with one of these words and 1 by the other.

Theorem 1. *If a transducer $\mathcal{M}$ contains a covert channel, it can be implemented by transducers $\mathcal{E}(h, h_0, h_1, h')$ and $\mathcal{D}(\ell, \ell_0, \ell_1, \ell')$ as depicted in Fig. 6, where:*

- $h, h_0, h_1, h' \in H^*, \quad \ell, \ell_0, \ell_1, \ell' \in L^*$,
- $h_0 \cdot h_1 \neq h_1 \cdot h_0$ *and* $\ell_0 \cdot \ell_1 \neq \ell_1 \cdot \ell_0$.



(a) Transducer $\mathcal{E}(h, h_0, h_1, h')$ (b) Transducer $\mathcal{D}(\ell, \ell_0, \ell_1, \ell')$

Fig. 6. General form of encoder and decoder.

Proof. Suppose $\mathcal{M}$ contains a covert channel implemented by $\mathcal{E}_0$ and $\mathcal{D}_0$. Transducer $\mathcal{E}_0 = \langle S, s_0, \{0,1\}^* \times H^*, \Delta, F\rangle$ is complete on its input. Recall that every state of S is both reachable and co-reachable. By Lemma 5, there exist words $w, w_0, w_1, w' \in Dom(\mathcal{E}_0) = \{0,1\}^*$, with $w_0 \cdot w_1 \neq w_1 \cdot w_0$ such that there exist paths $s_0 \xRightarrow{w|v} s$, $s \xRightarrow{w_0|v_0} s$, and $s \xRightarrow{w_1|v_1} s$. In addition, there is a final state $s_f \in F$ and a run $s \xRightarrow{w'|v'} s_f$.

We claim that $v_0 \cdot v_1 \neq v_1 \cdot v_0$. By contradiction, suppose $v_0 \cdot v_1 = v_1 \cdot v_0$. Then $v \cdot v_0 \cdot v_1 \cdot v' = v \cdot v_1 \cdot v_0 \cdot v' \in \mathcal{E}_0(w \cdot w_0 \cdot w_1 \cdot w') \cap \mathcal{E}_0(w \cdot w_1 \cdot w_0 \cdot w')$. We can choose $\mathcal{D}_0 \circ \mathcal{M}$ complete on $Im(\mathcal{E})$, because we can replace $\mathcal{E}_0$ by $\mathcal{E}_0' = Id_0(Dom(\mathcal{D}_0 \circ \mathcal{M})) \circ \mathcal{E}_0$. Thus $(\mathcal{D}_0 \circ \mathcal{M})(v \cdot v_0 \cdot v_1 \cdot v') = w \cdot w_0 \cdot w_1 \cdot w'$ because $\mathcal{D}_0 \circ \mathcal{M} \circ \mathcal{E}_0 = Id(\{0,1\}^*)$. The same reasoning on $v \cdot v_1 \cdot v_0 \cdot v'$ yields $(\mathcal{D}_0 \circ \mathcal{M})(v \cdot v_1 \cdot v_0 \cdot v') = (\mathcal{D}_0 \circ \mathcal{M})(v \cdot v_0 \cdot v_1 \cdot v') = w \cdot w_1 \cdot w_0 \cdot w'$, which is a contradiction with $w_0 \cdot w_1 \neq w_1 \cdot w_0$.

Let $\mathcal{E}_1 = \mathcal{E}(v, v_0, v_1, v')$ and $\mathcal{F} = \mathcal{E}(w, w_0, w_1, w')$ having the structure depicted in Fig. 6(a). We can see that $\mathcal{E}_1 \subseteq \mathcal{E}_0 \circ \mathcal{F}$. Let $\mathcal{G} = \mathcal{D}(w, w_0, w_1, w')$ and $\mathcal{D}_0' = \mathcal{G} \circ \mathcal{D}_0$, having the structure depicted in Fig. 6(b). We have

$$\mathcal{D}_0' \circ \mathcal{M} \circ \mathcal{E}_1 \subseteq \mathcal{G} \circ \mathcal{D}_0 \circ \mathcal{M} \circ \mathcal{E}_0 \circ \mathcal{F} = \mathcal{G} \circ \mathcal{F} = Id(\{0,1\}^*)$$

Let $v \in \{0,1\}^*$. Either $(\mathcal{D}_0' \circ \mathcal{M} \circ \mathcal{E}_1)(w) = w$ or $(\mathcal{D}_0' \circ \mathcal{M} \circ \mathcal{E}_1)(w) = \emptyset$. Since $Im(\mathcal{E}_1) \subseteq Im(\mathcal{E}_0) \subseteq Dom(\mathcal{D}_0 \circ \mathcal{M})$, $(\mathcal{D}_0' \circ \mathcal{M} \circ \mathcal{E}_1)(w)$ cannot be empty so $(\mathcal{D}_0' \circ \mathcal{M} \circ \mathcal{E}_1)(v) = v$. Therefore $\mathcal{E}_1$ is an encoder for $\mathcal{M}$.

Transducer $\mathcal{M}' = Id(Dom(\mathcal{D}_0')) \circ \mathcal{M} \circ \mathcal{E}_1 = \langle S', s_0', \{0,1\}^* \times L^*, \Delta', F'\rangle$ is complete on $\{0,1\}^*$. By Lemma 5, there exist states $s' \in S'$, $s_f' \in F'$, words $u, u_0, u_1, u' \in Dom(\mathcal{M}') \subseteq \{0,1\}^*$, with $u_0 \cdot u_1 \neq u_1 \cdot u_0$, and runs $s_0' \xRightarrow{u|\ell} s'$, $s' \xRightarrow{u_0|\ell_0} s'$, $s' \xRightarrow{u_1|\ell_1} s'$, and $s' \xRightarrow{u'|\ell'} s_f'$. Let $\mathcal{D} = \mathcal{D}(\ell, \ell_0, \ell_1, \ell')$ as depicted in Fig. 6(b). We have $\mathcal{D} \circ \mathcal{M}' \circ \mathcal{F}' = Id(\{0,1\}^*)$ with $\mathcal{F}' =$

$\mathcal{E}(u, u_0, u_1, u')$. We have

$$\mathcal{D} \circ \mathcal{M}' \circ \mathcal{F}' = \mathcal{D}' \circ Id(Dom(\mathcal{D}'_0)) \circ \mathcal{M} \circ \mathcal{E}_1 \circ \mathcal{F}'.$$

Since $Dom(\mathcal{D}) \subseteq Dom(\mathcal{D}'_0)$, we obtain $\mathcal{D} \circ Id(Dom(\mathcal{D}'_0)) = \mathcal{D}$.

Encoder $\mathcal{E} = \mathcal{E}_1 \circ \mathcal{F}'$ can be put under the form depicted in Fig. 6(a), by defining h, h_0, h_1, h' as follows. Suppose $u = u^1 \cdots u^j$, $u_0 = u_0^1 \cdots u_0^k$, $u_1 = u_1^1 \cdots u_1^m$, and $u' = u'^1 \cdots u'^m$, where each $u^i \in \{0, 1\}$. Then let $h = v \cdot v_{u^1} \cdots v_{u^j}$, $h_0 = v_{u_0^1} \cdots v_{u_0^k}$, $h_1 = v_{u_1^1} \cdots v_{u_1^n}$, $h' = v_{u'^1} \cdots v_{u'^m} \cdot v'$. Suppose that h_1 and h_2 commute. Then

$$v_{u_0^1} \cdots v_{u_0^k} \cdot v_{u_1^1} \cdots v_{u_1^n} = v_{u_1^1} \cdots v_{u_1^n} \cdot v_{u_0^1} \cdots v_{u_0^k}.$$

Since v_0 and v_1 form a code, this can happen only if the sequences of indexes are the same. That is to say if $u_0 \cdot u_1 = u_1 \cdot u_0$, which is a contradiction. Therefore h_0 and h_1 do not commute.

The property $\ell_0 \cdot \ell_1 \neq \ell_1 \cdot \ell_0$ can be proved in similar fashion as $v_0 \cdot v_1 \neq v_1 \cdot v_0$ has been, which concludes the proof.

As a result, we obtain a necessary condition on the structure of encoders and decoders, which can be transposed onto the system itself.

Theorem 2. *Let $\mathcal{M} = \langle S, s_0, H^* \times L^*, \Delta, F \rangle$ be a transducer that contains a covert channel. Then there exist a state $s \in S$ and four words $h_0, h_1 \in H^*$ and $\ell_0, \ell_1 \in L^*$ such that $s \xRightarrow{h_0|\ell_0} s$ and $s \xRightarrow{h_1|\ell_1} s$. Moreover, $h_1 \cdot h_0 \neq h_0 \cdot h_1$ and $\ell_1 \cdot \ell_0 \neq \ell_0 \cdot \ell_1$.*

Proof. Since there is a covert channel in $\mathcal{M}$, $\mathcal{E}(h, h_0, h_1, h')$ and $\mathcal{D}(\ell, \ell_0, \ell_1, \ell')$ implement a covert channel for $\mathcal{M}$, by Theorem 1. So, $\{(h|\ell) \cdot \{(h_0|\ell_0), (h_1|\ell_1)\}^* \cdot (h'|\ell')\} \subseteq \mathcal{L}(\mathcal{M})$. As $(h_0|\ell_0) \cdot (h_1|\ell_1) \neq (h_1|\ell_1) \cdot (h_0|\ell_0)$, by Lemma 4, there is a state $s \in S$ and four words $h'_0, h'_1 \in H^*$ and $\ell'_0, \ell'_1 \in L^*$ such that $s \xRightarrow{h'_0|\ell'_0} s$, $s \xRightarrow{h'_1|\ell'_1} s$, $h'_1 \cdot h'_0 \neq h'_0 \cdot h'_1$, and $\ell'_1 \cdot \ell'_0 \neq \ell'_0 \cdot \ell'_1$.

Such a state s in a system exhibits a behaviour similar to an *encoding node* in [21], so we call it an *encoding state*.

We can now prove the claim of Section 2 stating that transducer $\mathcal{M}_2$ depicted in Fig. 3 does not contain a covert channel. Since $Dom(\mathcal{M}_2) = \{h\}^*$, there are no words h_0 and h_1 in $Dom(\mathcal{M}_2)$ such that $h_0 \cdot h_1 \neq h_1 \cdot h_0$ which contradicts the necessary condition of the previous theorem.

It should be noted that the presence of such an encoding state does not however guarantee the existence of a covert channel in the general case. For example, in the (non functional) system of Fig. 7, state s_4 is

an encoding state. However, an h can also lead to s_3, which simulates s_4, but in which no word can be encoded: indeed, after a h_0 or a h_1, both a ℓ_0 and a ℓ_1 can be produced. Hence, all words made of h_0 and h_1 of a given length n will have the same set of images. More precisely, for any word $u = h \cdot h_{i_1} \cdots h_{i_n}$, where $\forall k, i_k \in \{0, 1\}$, $\mathcal{N}(u) = \ell \cdot (\ell_0 + \ell_1)^n$. In that case, the non-functionality of $\mathcal{N}$ breaks the locality of the encoding state property.

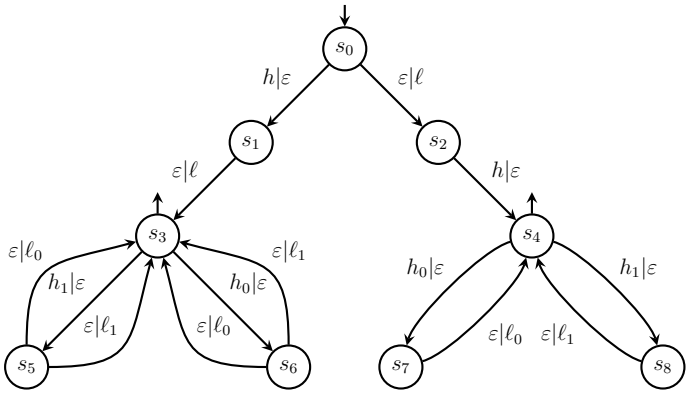


Fig. 7. Transducer $\mathcal{N}$ with an encoding state q_4 but no covert channel

5 Covert channels cannot be detected

We shall now prove that, unfortunately, the existence of a covert channel is undecidable in the general case of our setting. This is done by reducing Post's Correspondence Problem (PCP) [27]: starting from an instance $\mathcal{I}$ of PCP, we build a transducer $\mathcal{M}_{\mathcal{I}}$ such that $\mathcal{I}$ has a solution if and only if there exist an encoder and a decoder implementing a covert channel for $\mathcal{M}_{\mathcal{I}}$. The construction builds on the undecidability proof for transducer equality [28], with an involved additional construction to obtain the channel property. The main idea underlying the transducer construction is to require a sequence of indexes along to a bit, and transmit the correct bit if and only if the sequence is a non-trivial solution of the instance $\mathcal{I}$. Otherwise the system can insert errors.

Theorem 3. *The problem of existence of a covert channel in a transducer is undecidable.*

We first give the construction of transducer $\mathcal{M}_{\mathcal{I}}$ from an instance $\mathcal{I}$ of PCP and we then give two lemmas proving the correctness of this construction. Let $\mathcal{I} = \langle (x_1, y_1) \dots, (x_n, y_n) \rangle \in ((A^*)^2)^n$ be an instance of PCP over alphabet A. We consider the alphabets $B = \{\top, \bot\}$, $N = \{1, \dots, n\}$, $A_B = A \cup B$, and $N_B = N \cup B$. For $b \in B$, we define $\overline{b}$ by $\overline{\top} = \bot$ and $\overline{\bot} = \top$. Recall that such an instance can also be seen as a pair of morphisms x and y, with $x(\sigma) = x_{i_1} \cdots x_{i_k}$ and $y(\sigma) = y_{i_1} \cdots y_{i_k}$ for any word $\sigma = i_1 \cdots i_k \in N^*$. Hence PCP can be reformulated as the existence of a sequence σ, with $|\sigma| > 0$, such that $x(\sigma) = y(\sigma)$.

We now build a transducer $\mathcal{M}_{\mathcal{I}}$ in $N_B^* \times A_B^*$ which computes a relation such that for $b \in B$ and $\sigma \in N^*$:

$$\mathcal{M}_{\mathcal{I}}(b \cdot \sigma) = \left(A^+ \cdot b\right) \cup \left((A^+ \setminus \{x(\sigma)\}) \cdot \overline{b}\right) \cup \left((A^+ \setminus \{y(\sigma)\}) \cdot \overline{b}\right)$$

This transducer takes as an input a bit and a sequence of indexes and outputs

- either any non-empty word followed by the same bit
- or a word which is not the image of the sequence by x followed by the opposite of the input bit
- or a word which is not the image of the sequence by y followed by the opposite of the input bit

This relation is extended to N_B^* by $\mathcal{M}_{\mathcal{I}}(\varepsilon) = \{\varepsilon\}$ and for $b_1, \dots, b_p \in B$ and $\sigma_1, \dots, \sigma_p \in N^*$:

$$\mathcal{M}_{\mathcal{I}}(b_1 \cdot \sigma_1 \cdots b_p \cdot \sigma_p) = \mathcal{M}_{\mathcal{I}}(b_1 \cdot \sigma_1) \cdots \mathcal{M}_{\mathcal{I}}(b_p \cdot \sigma_p)$$

while $\mathcal{M}_{\mathcal{I}}(v) = \emptyset$ if $v \notin (B \cdot N^*)^*$.

The construction of $\mathcal{M}_{\mathcal{I}} = \langle Q, q_0, N_B^* \times A_B^*, \Delta, \{q_0\} \rangle$ is as follows. The set Q of states of $\mathcal{M}_{\mathcal{I}}$ is:

$$Q = \{q_0\} \cup B \times (\{q_*, q_x, q_y, q_>, q_<, q_\neq\} \cup Q_{\mathcal{I}})$$

where

$$Q_{\mathcal{I}} = \left(\bigcup_{i=1}^{n} \bigcup_{j=1}^{|x_i|} \{q_x^{i,j}\} \right) \cup \left(\bigcup_{i=1}^{n} \bigcup_{j=1}^{|y_i|} \{q_y^{i,j}\} \right)$$

is a set containing a state for each letter in each word of the instance $\mathcal{I}$. The only initial and final state is q_0. The set Δ of transitions of $\mathcal{M}_{\mathcal{I}}$ is built by the following rules, for each $b \in B$, $z \in \{x, y\}$, $i \in N$, and $a \in A$:

(R1) For $q \in \{q_*, q_x, q_y\}$, $q_0 \xrightarrow{b|\varepsilon} (b, q) \in \Delta$; $\mathcal{M}_{\mathcal{I}}$ reads b and make the initial nondeterministic choice of outputting

– either any non-empty word followed by b (state q_*),
– or a word which is not $x(\sigma)$ followed by $\overline{b}$ (state q_x),
– or a word which is not $y(\sigma)$ followed by $\overline{b}$ (state q_y).

(R2) $(b, q_*) \xrightarrow{i|\varepsilon} (b, q_*) \in \Delta$,

(R3) $(b, q_*) \xrightarrow{\varepsilon|a} (b, q_*) \in \Delta$; this rule and the previous simply allow the state q_* to read and write anything (reading on N and writing on A).

(R4) $(b, q_*) \xrightarrow{\varepsilon|a\cdot b} q_0 \in \Delta$; $\mathcal{M}_\mathcal{I}$ outputs a letter (to prevent the output of an empty word) and the bit that was read on the transition from q_0.

(R5) If $|z_i| > 0$, then $(b, q_z) \xrightarrow{i|\varepsilon} (b, q_z^{i,1}) \in \Delta$,

(R6) For $1 \leq j < |z_i|$, $(b, q_z^{i,j}) \xrightarrow{\varepsilon|z_i[j]} (b, q_z^{i,j+1}) \in \Delta$,

(R7) $(b, q_z^{i,|z_i|}) \xrightarrow{\varepsilon|z_i[|z_i|]} (b, q_z) \in \Delta$; the transitions created by the three last rules enable $\mathcal{M}_\mathcal{I}$ to read an input i and produce z_i, going back to q_z at the end.

(R8) If $|z_i| = 0$, then $(b, q_z) \xrightarrow{i|\varepsilon} (b, q_z) \in \Delta$; this rule is analogous to rules (R5-7) when $z_i = \varepsilon$.

(R9) For $1 \leq j < |z_i|$, $(b, q_z^{i,j}) \xrightarrow{\varepsilon|\varepsilon} (b, q_<) \in \Delta$; these transitions stop the outputting of z_i, going to $q_<$.

(R10) $(b, q_<) \xrightarrow{i|\varepsilon} (b, q_<) \in \Delta$; in $q_<$, the input is read but no output is produced.

(R11) $(b, q_z) \xrightarrow{\varepsilon|a} (b, q_>) \in \Delta$,

(R12) $(b, q_>) \xrightarrow{\varepsilon|a} (b, q_>) \in \Delta$; these transitions output at least one letter of A without reading anymore input (*i.e.* it should have all been read before).

(R13) For $1 \leq j < |z_i|$, and if $a \neq z_i[j]$, $(b, q_z^{i,j}) \xrightarrow{\varepsilon|a} (b, q_\neq) \in \Delta$; taking these transitions introduce a wrong letter in z_i.

(R14) $(b, q_\neq) \xrightarrow{i|\varepsilon} (b, q_\neq) \in \Delta$,

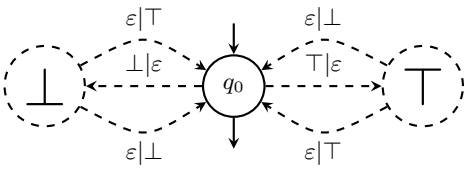
(R15) $(b, q_\neq) \xrightarrow{\varepsilon|a} (b, q_\neq) \in \Delta$; at state $q_\neq$, anything is read (on alphabet N) and anything is produced (on alphabet A)

(R16) For $q \in \{q_<, q_>, q_\neq\}$, $(b, q) \xrightarrow{\varepsilon|\overline{b}} q_0 \in \Delta$; returning from a state where an error has been made produces the opposite of the input bit.

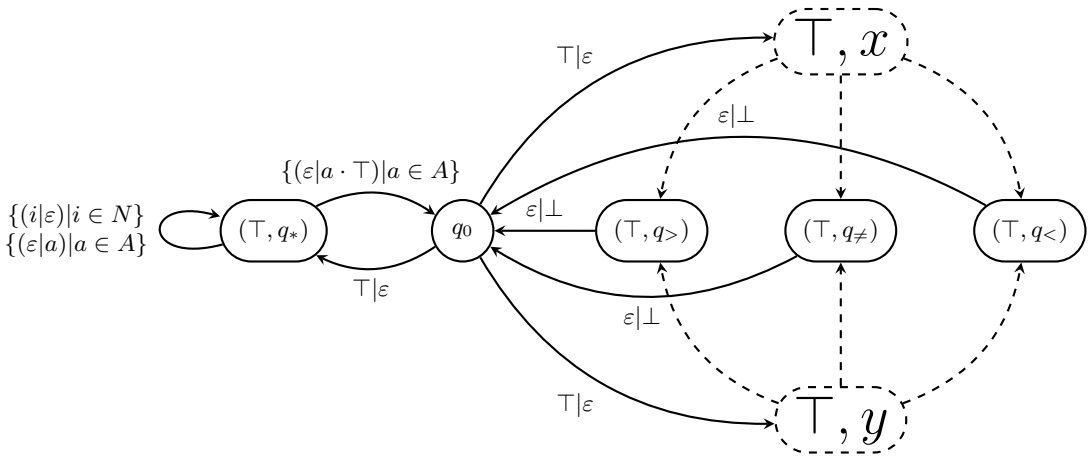
Transducer $\mathcal{M}_\mathcal{I}$ is composed of two symmetrical parts that keep in memory one bit b of information (see Fig. 8(a)). The part of $\mathcal{M}_\mathcal{I}$ consisting of state q_* does not look at its input and generates any word of A^+,

appending b after it. On the other hand, on input $b \cdot \sigma$ ($|\sigma| > 0$), the other states (which will be called the diff-part in the sequel) generate either a word which is not $x(\sigma)$, or a word which is not $y(\sigma)$, appending $\overline{b}$ after it (see Fig. 8(b)).

The parts of $\mathcal{M}_\mathcal{I}$ relative to x and y are similar, hence rules (R5-13) are presented in a general form for z, representing either x or y. Rules (R5-8) create a sub-part of $\mathcal{M}_\mathcal{I}$ able to produce $z(\sigma)$. However, since q_z is not an accepting state, this exact output shall not be produced in this part. Indeed, rules (R9-15) introduce errors in this word. These errors can be of three forms. Firstly, the outputted word can be a strict prefix of $z(\sigma)$ (rules (R9-10)). Secondly, the outputted word can contain $z(\sigma)$ as a strict prefix (rules (R11-12)). Thirdly, an error can be introduced by producing a letter than was not the one expected in an output of z_i (rules (R13-15)). The structure of this part of $\mathcal{M}_\mathcal{I}$ (for $b = \top$ and $z = x$) is depicted in Fig. 9. The transitions of rules (R4) and (R16) lead back to q_0, allowing the whole process to be repeated.



(a) Symmetry of $\top$ and $\perp$ in $\mathcal{M}_\mathcal{I}$



(b) Structure of the $\top$ part of $\mathcal{M}_\mathcal{I}$

Fig. 8. Global structure of $\mathcal{M}_\mathcal{I}$.

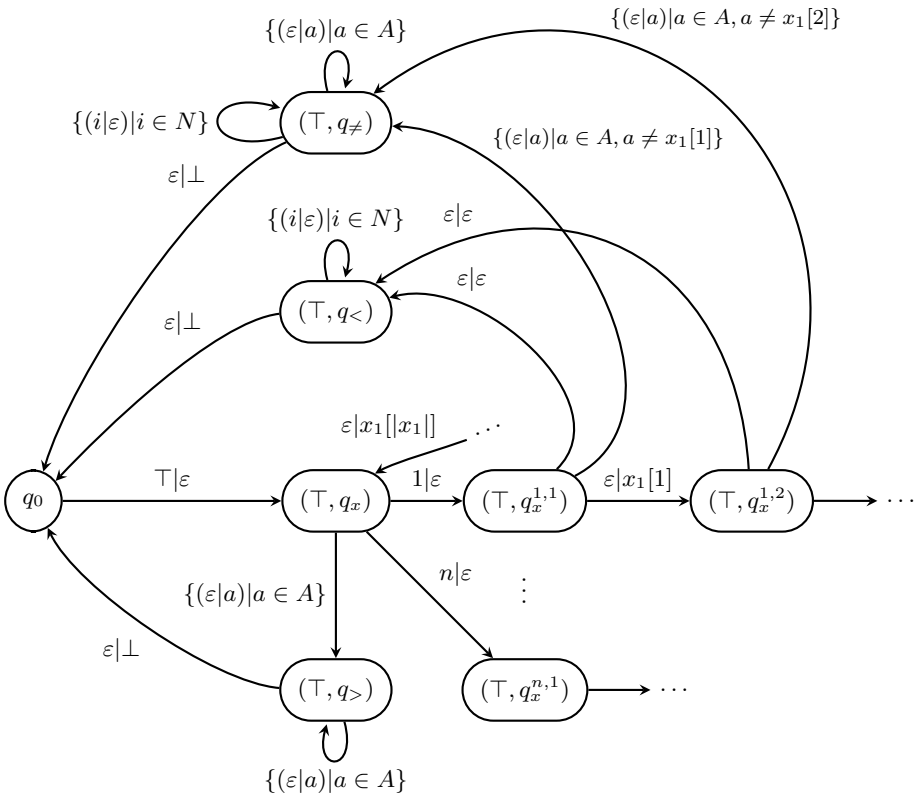


Fig. 9. Structure of the $(\top, x)$ part of $\mathcal{M}_{\mathcal{I}}$ that accepts $\{(\top \cdot \sigma, u \cdot \bot)|u \neq x(\sigma)\}^*$.

If, for an input $b \cdot \sigma$ with $|\sigma| > 0$, the sequence σ is a solution of $\mathcal{I}$, then $w = x(\sigma) = y(\sigma)$ will not be generated by the diff-part of $\mathcal{M}_{\mathcal{I}}$, hence $w \cdot b$ will be an output whereas $w \cdot \overline{b}$ will not[5]. Conversely, if the sequence σ is not a solution of $\mathcal{I}$, then $w = x(\sigma) \neq y(\sigma)$ will be generated by the diff-part of $\mathcal{M}_{\mathcal{I}}$ (in this case in the "y part" of the transducer), hence both $w \cdot b$ and $w \cdot \overline{b}$ will be outputs. Note that in both cases, there will be other outputs: all $u \cdot b$ and $u \cdot \overline{b}$ for $u \in A^+ \setminus \{w\}$. When $|\sigma| = 0$, which is always a trivial solution of $\mathcal{I}$, the empty word ε cannot be produced in the q_* part of $\mathcal{M}$. Hence neither b nor $\overline{b}$ will be produced (alone). Even if the above mentioned only the case of an input in $B \cdot N^*$, it can be generalized to the case of $(B \cdot N^*)^*$. Indeed, q_0 is the only initial and final state and the structure of $\mathcal{M}_{\mathcal{I}}$ ensures that q_0 is left reading a letter of

[5] We can assume that there is no index i such that $x_i = y_i = \varepsilon$, hence $w \neq \varepsilon$.

B, reached producing a (possibly different) letter of B, and that no other transition either reads or outputs a letter of B.

The two following lemmas prove the correctness of this construction.

Lemma 6. *If $\mathcal{I}$ has a solution then $\mathcal{M}_{\mathcal{I}}$ has a covert channel.*

Proof. Suppose $\sigma = i_1 \cdots i_k$ is a solution of $\mathcal{I}$, with $k > 0$ and $w = x(\sigma) = y(\sigma)$. Consider transducer $\mathcal{M}_\sigma$ of $\{0,1\}^* \times N_B^*$ that accepts the relation $\{(0, \top \cdot \sigma), (1, \bot \cdot \sigma)\}^*$, depicted in Fig. 10(a). Let $\beta_1 \cdots \beta_p$ be a word of $\{0,1\}^*$. Then its only image by $\mathcal{M}_\sigma$ is

$$v = b_1 \cdot \sigma \cdots b_p \cdot \sigma$$

where

$$\forall 1 \leq j \leq p,\ b_j = \top \text{ iff } \beta_j = 0 \text{ and } b_j = \bot \text{ iff } \beta_j = 1.$$

It is clear that, since σ is a solution of $\mathcal{I}$, the outputs of $\mathcal{M}_{\mathcal{I}}$ on input v will be

$$\{u_1 \cdot b'_1 \cdots u_p \cdot b'_p | \forall 1 \leq j \leq p, u_j \in A^+ \wedge b'_j \in B \wedge (u_j = w \Rightarrow b'_j = b_j)\}.$$

In particular, the word $v' = w \cdot b_1 \cdots w \cdot b_p$ is an output of $\mathcal{M}_{\mathcal{I}}$ on v, whereas no other word $w \cdot b'_1 \cdots w \cdot b'_p$ is whenever $(b_1, \ldots, b_p) \neq (b'_1, \ldots, b'_p)$. Let us consider transducer $\mathcal{M}_w$ of $A_B^* \times \{0,1\}^*$ that accepts the relation $\{(w \cdot \top, 0), (w \cdot \bot, 1)\}^*$, depicted in Fig. 10(b). Any input $w \cdot b'_1 \cdots w \cdot b'_p$ is transformed by $\mathcal{M}_w$ into $\beta'_1 \cdots \beta'_p$ where

$$\forall 1 \leq j \leq p,\ \beta'_j = 0 \text{ iff } b'_j = \top \text{ and } \beta'_j = 1 \text{ iff } b'_j = \bot.$$

Any other form of input will not be accepted by $\mathcal{M}_w$. For the particular input v', $\mathcal{M}_w$ will yield the original word $\beta_1 \cdots \beta_p$. Since v is the only output of $\mathcal{M}_{\mathcal{I}} \circ \mathcal{M}_\sigma$ that can be accepted by $\mathcal{M}_w$, we now have that

$$(\mathcal{M}_w \circ \mathcal{M}_{\mathcal{I}} \circ \mathcal{M}_\sigma)(\beta_1 \cdots \beta_p) = \beta_1 \cdots \beta_p$$

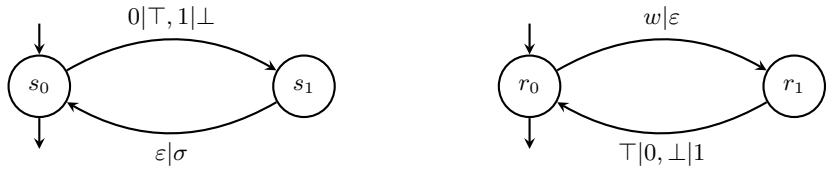
and therefore $\mathcal{M}_\sigma$ and $\mathcal{M}_w$ implement a covert channel for $\mathcal{M}$.

Lemma 7. *If $\mathcal{I}$ has no solution then $\mathcal{M}_{\mathcal{I}}$ has no covert channel.*

Proof. Suppose $\mathcal{I}$ has no trivial solution. Then $\mathcal{M}$ is the relation

$$\{(b \cdot \sigma, u \cdot b') | \sigma \in N^*, u \in A^+ \text{ and } b, b' \in B\}^*.$$

All words in $Dom(\mathcal{M}_{\mathcal{I}})$ with same number of letters of B have exactly the same set of images. Namely for $u \in N_B^*$, $\mathcal{M}_{\mathcal{I}}(u) = (A^+ \cdot B)^k$, where



(a) Transducer $\mathcal{M}_\sigma$ from $\{0,1\}^*$ to N_B^* (b) Transducer $\mathcal{M}_w$ from A_B^* to $\{0,1\}^*$

Fig. 10. Encoder and decoders $\mathcal{M}_\sigma$ and $\mathcal{M}_w$, where σ is a solution of the instance $\mathcal{I}$ of PCP and w the corresponding word.

$k = |u|_B$. Suppose there is an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$ that implement a covert channel for $\mathcal{M}_\mathcal{I}$. By Theorem 1, these transducers can be chosen as in Fig. 6. Let $\beta_1 \cdots \beta_p \in \{0,1\}^*$. We consider words

$$u = \beta_1 \cdots \beta_p \cdot \overline{\beta_1} \cdots \overline{\beta_p} \quad \text{and} \quad u' = \overline{\beta_1} \cdots \overline{\beta_p} \cdot \beta_1 \cdots \beta_p$$

where $\overline{0} = 1$ and $\overline{1} = 0$. The image of these words by $\mathcal{E}$ are respectively

$$v = h \cdot h_{\beta_1} \cdots h_{\beta_p} \cdot h_{\overline{\beta_1}} \cdots h_{\overline{\beta_p}} \cdot h' \quad \text{and} \quad v' = h \cdot h_{\overline{\beta_1}} \cdots h_{\overline{\beta_p}} \cdot h_{\beta_1} \cdots h_{\beta_p} \cdot h'.$$

Since the image of v (resp. v') by $\mathcal{D} \circ \mathcal{M}_\mathcal{I}$ is u (resp. u'), v and v' must have at least an image by $\mathcal{M}_\mathcal{I}$. Moreover, v and v' contain exactly the same number k of letter of B. Hence by $\mathcal{M}_\mathcal{I}$, both v and v' have the same set of images $(A^+ \cdot B)^k$. So the image of v and v' by $\mathcal{D} \circ \mathcal{M}_\mathcal{I}$ are the same. Therefore, $(\mathcal{D} \circ \mathcal{M}_\mathcal{I} \circ \mathcal{E})(u) = (\mathcal{D} \circ \mathcal{M}_\mathcal{I} \circ \mathcal{E})(u')$ while $u \neq u'$, which is a contradiction with the fact that $\mathcal{E}$ and $\mathcal{D}$ implement a covert channel for $\mathcal{M}_\mathcal{I}$.

We can thus conclude that $\mathcal{M}_\mathcal{I}$ has a covert channel if and only if $\mathcal{I}$ has a (non-trivial) solution. Since PCP is undecidable, the problem of existence of a covert channel in a transducer is undecidable.

Example 3. Consider the following instance of PCP:

$$\mathcal{I}_0 = \langle (abb, a), (b, abb), (a, bb) \rangle$$

The corresponding transducer $\mathcal{M}_{\mathcal{I}_0}$ is partly depicted (only the $\top, x$ part) in Figure 11. This instance has a solution $\sigma = 1311322$ which yields the word $w = abbaabbabbabb$. On input $\top 1311322$, $\mathcal{M}_{\mathcal{I}_0}$ can output any non-empty string followed by a $\top$ by a run looping through state q_*. In particular, $abbaabbabbabb\top$ is a possible output. On the same input, some other strings followed by a $\bot$ may be an output, *e.g.* $abbaabbabaa\bot$ which is the product of a run

$$q_0 \xrightarrow{\top|\varepsilon} q_x \xrightarrow{1|\varepsilon} q_x^{1,1} \xrightarrow{\varepsilon|a} q_x^{1,2} \xrightarrow{\varepsilon|b} q_x^{1,3} \xrightarrow{\varepsilon|b} q_x \xrightarrow{3|\varepsilon} q_x^{3,1} \xrightarrow{\varepsilon|a} q_x \xrightarrow{1|\varepsilon} q_x^{1,1} \xrightarrow{\varepsilon|a} \cdots$$

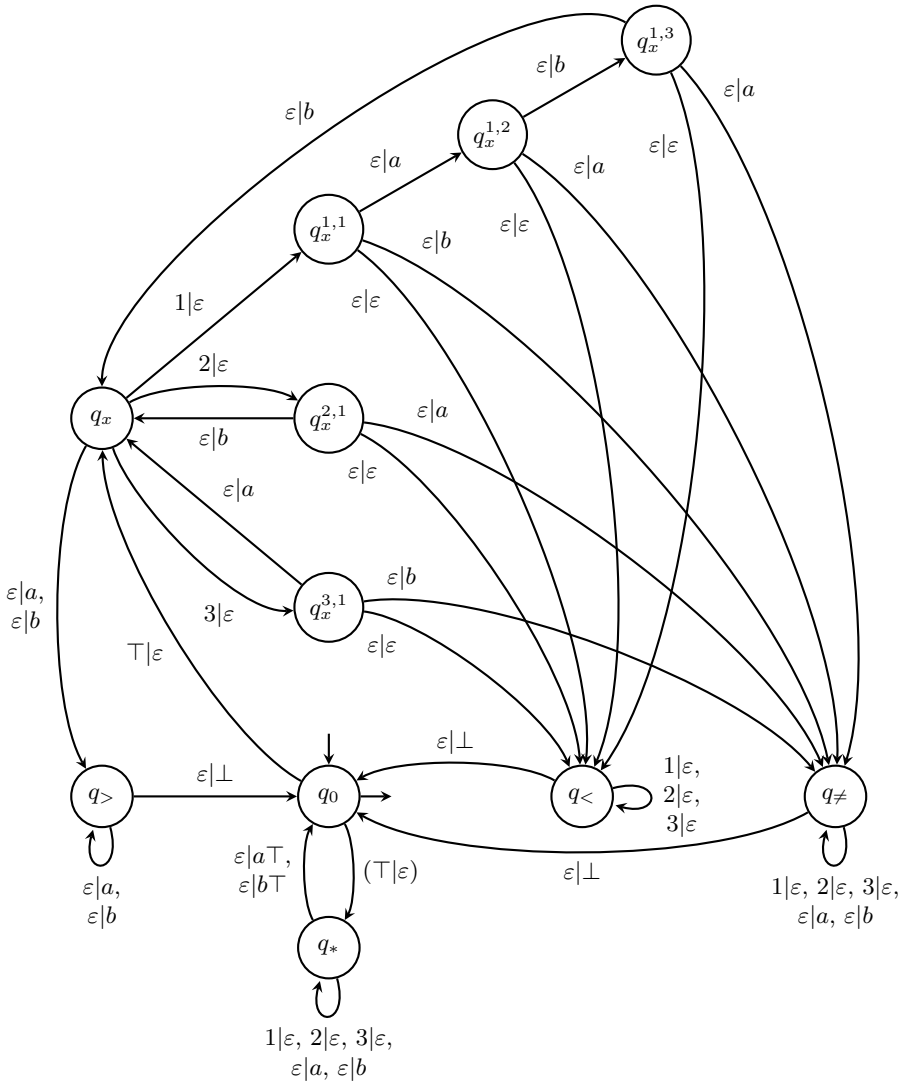


Fig. 11. Part of transducer $\mathcal{M}_{\mathcal{I}_0}$ encoding PCP instance $\mathcal{I}_0$. Only the $\top$-x quarter has been represented due to lack of space.

$$\cdots \xrightarrow{\varepsilon|a} q_x^{1,2} \xrightarrow{\varepsilon|b} q_x^{1,3} \xrightarrow{\varepsilon|b} q_x \xrightarrow{1|\varepsilon} q_x^{1,1} \xrightarrow{\varepsilon|a} q_x^{1,2} \xrightarrow{\varepsilon|b} q_x^{1,3} \xrightarrow{\varepsilon|a} q_{\neq} \xrightarrow{\varepsilon|a} q_{\neq} \xrightarrow{\varepsilon|\perp} q_0.$$

However, $abbaabbabbabb\perp$ is not an output, since after reading $\top 1311322$ and producing $abbaabbabbabb$, the run ends in state q_x (or q_y) which is not accepting and cannot reach q_0 without reading more input. Hence encoding 0 with $\top 1311322$ and 1 with $\perp 1311322$, while decoding 0 with $abbaabbabbabb\top$ and 1 with $abbaabbabbabb\perp$ yields a covert channel on $\mathcal{M}_{\mathcal{I}_0}$.

6 Covert channel synthesis for functional transducers

We finally show that covert channel synthesis is possible for functional transducers, with polynomial complexity. Intuitively, functional transducers introduce a small amount of noise in the system. Therefore, structural properties are sufficient to decide the existence of a covert channel.

Consider a transducer $\mathcal{M} = \langle S, s_0, H^* \times L^*, \Delta, F \rangle$ and a state $s \in S$. We define the transducer $\mathcal{M}_s = \langle S, s, H^* \times L^*, \Delta, \{s\} \rangle$ which differs from $\mathcal{M}$ only by its initial and final states.

Lemma 8. *Let $\mathcal{M} = \langle S, s_0, H^* \times L^*, \Delta, F \rangle$ be a functional transducer. Then $\forall s \in S$, $\mathcal{M}_s$ is also functional.*

Proof. Suppose there is a word w in the domain of $\mathcal{M}_s$ whose image contains at least two distinct words $\ell_0, \ell_1 \in L^*$. Consider the two corresponding runs $\rho_0 = s \xRightarrow{w|\ell_0} s$ and $\rho_1 = s \xRightarrow{w|\ell_1} s$ in $Runs(\mathcal{M}_s)$. As $s \in S = Reach(\mathcal{M}) \cap CoReach(\mathcal{M})$, there is a run $\rho = s_0 \xRightarrow{u|\ell} s \xRightarrow{v|\ell'} s_f \in Runs(\mathcal{M})$ with $s_f \in F$. So we can build two runs $\rho'_0 = s_0 \xRightarrow{u|\ell} s \xRightarrow{w|\ell_0} s \xRightarrow{v|\ell'} s_f$ and $\rho'_1 = s_0 \xRightarrow{u|\ell} s \xRightarrow{w|\ell_1} s \xRightarrow{v|\ell'} s_f$ in $Runs(\mathcal{M})$. Therefore, $\ell \cdot \ell_0 \cdot \ell'$ and $\ell \cdot \ell_1 \cdot \ell'$ are both images of $u \cdot w \cdot v$ and $\mathcal{M}$ is not functional, which is a contradiction.

Remark that, since s is both the initial and final state, and $\mathcal{M}_s$ is functional, we have $\forall w_0, w_1 \in Dom(\mathcal{M}_s)$, $\mathcal{M}_s(w_0 \cdot w_1) = \mathcal{M}_s(w_0) \cdot \mathcal{M}_s(w_1)$.

In the sequel, we call $\mathcal{E}(h, h_0, h_1, h')$ and $\mathcal{D}(\ell, \ell_0, \ell_1, \ell')$ (or $\mathcal{E}$ and $\mathcal{D}$ for short when their parameters are clear from the context) the two transducers depicted in Fig. 6. We also call $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ (or $\mathcal{E}_0$ and $\mathcal{D}_0$ for short when their parameters are clear from the context) the two transducers depicted in Fig. 12.

The following lemma expresses the fact that in the case of functional transducers, the existence of a covert channel is equivalent to the existence of an encoding state, which Theorem 2 established as a necessary condition.

(a) Transducer $\mathcal{E}_0(h_0, h_1)$ (b) Transducer $\mathcal{D}_0(\ell_0, \ell_1)$

Fig. 12. General form of encoder and decoder.

Lemma 9. *Let $\mathcal{M} = \langle S, s_0, H^* \times L^*, \Delta, F\rangle$ be a functional transducer. There is a covert channel in $\mathcal{M}$ if and only if there exist $s \in S$, $h_0, h_1 \in H^*$ and $\ell_0, \ell_1 \in L^*$ such that the transducers $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ implement a covert channel for $\mathcal{M}_s$.*

Proof. Suppose $\mathcal{M}$ contains a covert channel. Then by Theorem 2, there is a state s at the intersection of two cycles: $s \xRightarrow{h_0|\ell_0} s$ and $s \xRightarrow{h_1|\ell_1} s$. In $\mathcal{M}_s$, which is functional, s is both the initial and final state. Therefore for any $u, v \in \mathit{Dom}(\mathcal{M}_s)$, $\mathcal{M}_s(u) \cdot \mathcal{M}_s(v) = \mathcal{M}_s(u \cdot v)$. In particular, it is true for any word u in the language $(h_0 + h_1)^* \subseteq \mathit{Dom}(\mathcal{M}_s)$. If $u = h_{b_1} \cdots h_{b_n}$, where $b_1, \ldots, b_n \in \{0, 1\}$, then

$$v = \mathcal{M}_s(u) = \mathcal{M}_s(h_{b_1}) \cdots \mathcal{M}_s(h_{b_n}) = \ell_{b_1} \cdots \ell_{b_n}.$$

Remark that the decomposition of u, and hence of v, is unique since $h_1 \cdot h_0 \neq h_0 \cdot h_1$ and $\ell_1 \cdot \ell_0 \neq \ell_0 \cdot \ell_1$. It is therefore clear that $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ implement a covert channel for $\mathcal{M}_s$.

On the other hand, suppose there is a state $s \in S$, and two transducers $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ that implement a covert channel for $\mathcal{M}_s$. Since s is both reachable and coreachable, there exist some runs $s_0 \xRightarrow{(h|\ell)} s$ and $s \xRightarrow{(h'|\ell')} s_f$, with $s_f \in F$, $h, h' \in H$ and $\ell, \ell' \in L$. For any word $x \in (h_0 + h_1)^*$, $\mathcal{M}(h \cdot x \cdot h') = \ell \cdot \mathcal{M}_s(x) \cdot \ell'$ (since both $\mathcal{M}$ and $\mathcal{M}_s$ are functional). For a word $u = b_1 \cdots b_n \in \{0, 1\}^*$, let $v = \mathcal{E}(h, h_0, h_1, h')(u) = h \cdot h_{b_1} \cdots h_{b_n} \cdot h'$. Now let $w = \mathcal{M}(v) = \ell \cdot \ell_{b_1} \cdots \ell_{b_n} \cdot \ell'$. We have that $\mathcal{D}(\ell, \ell_0, \ell_1, \ell')(w) = b_1 \cdots b_n = u$. Therefore $\mathcal{E}(h, h_0, h_1, h')$ and $\mathcal{D}(\ell, \ell_0, \ell_1, \ell')$ implement a covert channel for $\mathcal{M}$.

In order to find encoding states, for any word $h \in H^*$, we consider the set $\mathit{NCI}(h, \mathcal{M})$ of words whose image by $\mathcal{M}$ do not commute with the image of h. More formally, given a transducer $\mathcal{M}$ and a word $h \in \mathit{Dom}(\mathcal{M})$,

we define the language $NCI(h, \mathcal{M}) = \{h' \in H^* \mid \mathcal{M}(h) \cdot \mathcal{M}(h') \neq \mathcal{M}(h') \cdot \mathcal{M}(h)\}$.

Lemma 10. *Given a functional transducer $\mathcal{M}$ and a word $h \in Dom(\mathcal{M})$, $NCI(h, \mathcal{M})$ is a rational subset of L^*.*

Proof. Let $\ell = \mathcal{M}(h)$. Consider the language $C(\ell) = \{\ell' \in L^* | \ell \cdot \ell' = \ell' \cdot \ell\}$ of the words commuting with ℓ. By Lemma 1, there exists a word $v \in L^*$ such that $C(\ell) = v^*$, therefore $C(\ell)$ is a rational language. Let $C'(\ell) = \{\ell' \in Im(\mathcal{M}) \mid \ell \cdot \ell' \neq \ell' \cdot \ell\} = Im(\mathcal{M}) \setminus C(\ell)$. Since the image of a rational transducer is rational, $Im(\mathcal{M})$ is rational and $C'(\ell) = Im(\mathcal{M}) \cap \overline{C(\ell)}$ is too. It is clear that $NCI(h, \mathcal{M}) = \mathcal{M}^{-1}(C'(\ell))$. Hence $NCI(h, \mathcal{M})$ is the inverse image by a rational transducer of a rational set, and is therefore rational.

We shall now prove that for a given state, it can be decided whether it is encoding.

Lemma 11. *Let $\mathcal{M} = \langle S, s, H^* \times L^*, \Delta, \{s\}\rangle$ be a functional transducer. For all $h \in \mathcal{M}^{-1}(Im(\mathcal{M}) \setminus \{\varepsilon\})$, there exist $h_0, h_1 \in H^*$, $\ell_0, \ell_1 \in L^*$ such that $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ implement a covert channel for $\mathcal{M}$ if and only if $NCI(h, \mathcal{M}) \neq \emptyset$.*

Proof. Remark that, by construction of $\mathcal{M}$ the initial state is the only final state. Let $h \in \mathcal{M}^{-1}(Im(\mathcal{M}) \setminus \{\varepsilon\})$. Also note that since $\mathcal{M}$ is functional and $\mathcal{M}(\varepsilon) = \varepsilon$, $h \neq \varepsilon$.

Suppose that $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ implement a covert channel for $\mathcal{M}$, and that $NCI(h, \mathcal{M}) = \emptyset$. For $i \in \{0, 1\}$, let $u_i = \mathcal{M}(h_i)$, and $u = \mathcal{M}(h)$. Since $h_0 \notin NCI(h, \mathcal{M})$ and $h_1 \notin NCI(h, \mathcal{M})$, $u \cdot u_0 = u_0 \cdot u$ and $u \cdot u_1 = u_1 \cdot u$. Then there exists $v \in L^*$ and three integers m, m_0, m_1 such that $u = v^m$, $u_0 = v^{m_0}$, $u_1 = v^{m_1}$ (by Lemma 1). Therefore $u_1 \cdot u_0 = u_0 \cdot u_1 = v^{n+p}$, which contradicts the fact that $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ implement a covert channel for $\mathcal{M}$.

Conversely, suppose that $NCI(h, \mathcal{M}) \neq \emptyset$. Let $h_0 = h$, $h_1 \in NCI(h, \mathcal{M})$, $\ell_0 = \mathcal{M}(h_0)$, and $\ell_1 = \mathcal{M}(h_1)$. Let $b = b_1 \cdots b_n$ be a word of $\{0, 1\}^*$. Then $\mathcal{E}_0(b) = h_{b_1} \cdots h_{b_n}$. Since s is both the initial and final state, and $\mathcal{M}$ is functional, $\forall w_0, w_1 \in Dom(\mathcal{M})$, $\mathcal{M}(w_0) \cdot \mathcal{M}(w_1) = \mathcal{M}(w_0 \cdot w_1)$. Therefore the image of $\mathcal{E}_0(b)$ by $\mathcal{M}$ is

$$(\mathcal{M} \circ \mathcal{E}_0)(b) = \mathcal{M}(h_{b_1} \cdots h_{b_n}) = \mathcal{M}(h_{b_1}) \cdots \mathcal{M}(h_{b_n}) = \ell_{b_1} \cdots \ell_{b_n}.$$

And, as $\ell_0 \cdot \ell_1 \neq \ell_1 \cdot \ell_0$ by construction of $NCI(h, \mathcal{M})$, $\ell_{b_1} \cdots \ell_{b_n} = \ell_{b'_1} \cdots \ell_{b'_n}$ iff $b_i = b'_i$, $\forall 1 \leq i \leq n$. Hence, $(\mathcal{D}_0 \circ \mathcal{M} \circ \mathcal{E}_0)(b) = b$. So $\mathcal{E}_0(h_0, h_1)$ and $\mathcal{D}_0(\ell_0, \ell_1)$ implement a covert channel for $\mathcal{M}$.

We shall now deduce the main result of this section, namely the decidability of the existence of covert channels in functional systems:

Theorem 4. *Let $\mathcal{M} = \langle S, s_0, H^* \times L^*, \Delta, F\rangle$ be a functional transducer. It can be decided in PTIME whether $\mathcal{M}$ has a covert channel. Moreover, an encoder and a decoder can be synthesized, if they exist.*

Proof. The decision procedure goes as follows: for each state $s \in S$, consider transducer $\mathcal{M}_s$. As before, $\mathcal{M}_s$ can be pruned so that all its states, in the set S_s, are both reachable and co-reachable. Then compute a word h whose image by $\mathcal{M}_s$ is not ε. This can be done by looking if there is $s_1, s_2 \in S_s$, s.t $s_1 \xrightarrow{h_e|\ell_e} s_2$ with $h_e \in H^*$ and $\ell_e \in L^+$ and finding a run $\rho = s_0 \Rightarrow s_1 \xrightarrow{h_e|\ell_e} s_2 \Rightarrow s_0$. If no such state can be found, then $Im(\mathcal{M}_s) = \{\varepsilon\}$ and it is clear that $\mathcal{M}_s$ does not have a covert channel. Computing S_s (the pruning of $\mathcal{M}_s$) can be done in $O(|\mathcal{M}|^2)$. The run ρ can be found from s_1 and s_2 in $O(|\mathcal{M}|^2)$ too. So computing h whose image by $\mathcal{M}_s$ is not ε can be done in $O(|\mathcal{M}|^2)$. Let $\ell = \mathcal{M}_s(h)$. Let $C(\ell) \subseteq L^*$ be the set of words that commute with ℓ. This set is v^* were v is the shortest word such that there exists $k \in \mathbb{N}$ such that $v^k = l$ (by Lemma 1). A deterministic automaton $\mathcal{A}_v$ of size $O(|v|)$ recognizes $\overline{v^*}$. An automaton $\mathcal{A}_{Im(\mathcal{M}_s)}$ of size $O(|\mathcal{M}|)$ recognizes $Im(\mathcal{M}_s)$. Therefore the intersection automaton of $\mathcal{A}_v$ and $\mathcal{A}_{Im(\mathcal{M}_s)}$, automaton $\mathcal{A}_{C'}$ of size $O(|v| \times |\mathcal{M}|)$ recognizes $C'(\ell) = Im(\mathcal{M}_s) \setminus v^*$. The emptiness problem for this automaton can be solved in $O((|v| \times |\mathcal{M}_s|)^3)$. If $C'(\ell)$ is empty, then so is its preimage by $\mathcal{M}$, and therefore $NCI(h, \mathcal{M}) = \emptyset$ and there is no covert channel (by Lemma 11). Otherwise, since $C'(\ell) \subseteq Im(\mathcal{M}_s)$, $\mathcal{M}_s^{-1}(C'(\ell)) = NCI(h, \mathcal{M}_s) \neq \emptyset$, and there is a covert channel in $\mathcal{M}_s$, which can be synthesized by the construction in the proof of Lemma 11, in linear time with respect to $|\mathcal{M}_s|$.

By Lemma 9, the existence of a covert channel in one transducer $\mathcal{M}_s$ is equivalent to the existence of a covert channel in $\mathcal{M}$, and the construction of the encoder and decoder for $\mathcal{M}$ from the ones for $\mathcal{M}_s$ can be done as in the proof of Lemma 9, in linear time with respect to $|\mathcal{M}_s|$.

Since $|v| \leq |\ell| \leq |\mathcal{M}_s| \leq |\mathcal{M}|$, the whole procedure goes in $O(|\mathcal{M}|) \times O(|\mathcal{M}|^2 + |\mathcal{M}|^6 + |\mathcal{M}|) = O(|\mathcal{M}|^7)$.

7 Conclusion and Future Work

In this work, we proposed a new definition for covert channels, based on transducer composition which significantly differs from the one based on iterated interference.

However, the existence problem itself is undecidable in the general case. But, in the case of functional transducers, the problem is decidable in polynomial time. The huge complexity gap suggests that for some subclass of transducers more general than functional ones, some decidability results may be obtained. We also need to extend our definition, in order to deal with cases where the high-level user can see a part of the low-level actions. Another direction for future work would be to investigate the control problem: can we find a controller to avoid covert channels in a system ? An orthogonal problem would be to extend this notion of covert channel to the framework of timed systems.

References

1. Z. Trabelsi, H. El Sayed, L. Frikha, T. Rabie, A novel covert channel based on the IP header record route option, Int. J. Adv. Media Commun. 1 (4) (2007) 328–350.
2. P. C. Kocher, Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems, in: Proc. CRYPTO'96, Springer-Verlag, 1996, pp. 104–113.
3. P. Kocher, J. Jaffe, B. Jun, Differential power analysis, in: Proc. CRYPTO'99, Vol. 1666 of LNCS, Springer-Verlag, 1999, pp. 388–397.
4. J.-J. Quisquater, D. Samyde, ElectroMagnetic Analysis (EMA): Measures and Couter-Measures for Smard Cards, in: Smart Card Programming and Security (E-smart 2001), Springer, 2001, pp. 200–210.
5. B. Lampson, A note on the confinement problem, Commun. ACM 16 (10) (1973) 613–615.
6. D. E. Bell, L. J. Lapadula, Secure computer systems: mathematical foundations, Tech. Rep. 2547, MITRE (1973).
7. J. Goguen, J. Meseguer, Security policy and security models, in: Proc.of ieee symposium on security and privacy, IEEE Computer Society Press, 1982, pp. 11–20.
8. R. Focardi, R. Gorrieri, Classification of security properties (part I: information flow), in: Foundations of security analysis and design I: FOSAD 2000 tutorial lectures, Vol. 2171 of LNCS, Springer-Verlag, 2001, pp. 331–396.
9. G. Lowe, Quantifying information flow, in: Proc. 15th IEEE workshop on computer security foundations (CSFW'02), IEEE Computer Society, Washington, DC, USA, 2002, p. 18.
10. I. S. Moskowitz, M. H. Kang, Covert Channels - Here to Stay?, in: Proc. COMPASS'94, IEEE Press, 1994, pp. 235–243.
11. J. Millen, Finite-state noiseless covert channels, in: Proceedings of the computer security foundations workshop II, 1989., 1989, pp. 81–86.
12. P. Malacaria, Assessing security threats of looping constructs, in: POPL '07: Proceedings of the 34th annual ACM SIGPLAN-SIGACT symposium on Principles of programming languages, ACM, New York, NY, USA, 2007, pp. 225–235.
13. B. Köpf, D. Basin, An information-theoretic model for adaptive side-channel attacks, in: Proc. 14th ACM Conf. on Computer and communications security (CCS'07), ACM, New York, NY, USA, 2007, pp. 286–296.
14. D. Sutherland, A model of information, in: Proc. of the 9th National Computer Security Conference, 1986.

15. J. Mullins, Nondeterministic admissible interference, Journal of Universal computer science 6 (11) (2000) 1054–1070.
16. P. Ryan, J. McLean, J. Millen, V. Gligor, Non-interference, who needs it?, in: Proc. 14th IEEE Computer Security Foundations Workshop, 2001, 2001, pp. 237–238.
17. L. Mazaré, Using Unification for Opacity Properties, in: Proc. of WITS, 2004, pp. 165–176.
18. J. Bryans, M. Koutny, P. Y. A. Ryan, Modelling Opacity using Petri Nets, Electronic Notes in Theoretical Computer Science 121 (2005) 101–115.
19. J. W. Bryans, M. Koutny, L. Mazaré, P. Y. A. Ryan, Opacity generalised to transition systems, International Journal of Information Security 7 (6) (2008) 421–435.
20. E. Badouel, M. Bednarczyk, A. Borzyszkowski, B. Caillaud, P. Darondeau, Concurrent secrets, Discrete Events Dynamic Systems 17 (4).
21. L. Hélouet, M. Zeitoun, A. Degorre, Scenarios and Covert channels: another game..., in: L. de Alfaro (Ed.), Proceedings of Games in Design and Verification (GDV'04), Vol. 119 of Electronic Notes in Theoretical Computer Science, Elsevier, 2005, pp. 93–116.
22. M. Lothaire, Combinatorics on words, Vol. 17 of Encyclopedia of Mathematics, Addison-Wesley, Reading, MA, 1983.
23. M. A. Harrison, Introduction to formal language theory, Addison-Wesley, 1978.
24. J. Sakarovitch, Éléments de théorie des automates, Vuibert Informatique, 2003.
25. C. C. Elgot, J. E. Mezei, On relations defined by generalized finite automata, IBM Journal Res. Develop. 9 (1965) 47–68.
26. S. Zdancewic, Challenges for Information-flow Security, in: Proceedings of the 1st International Workshop on the Programming Language Interference and Dependence (PLID'04), 2004.
27. E. L. Post, A variant of a recursively unsolvable problem, Bulletin of the American Mathematical Society 52 (4) (1946) 264–268.
28. E. Gurari, An introduction to the theory of computation, Computer Science Press, New York, NY, 1989.